International Journal of Humanoid Robotics

Domo: A Force Sensing Humanoid Robot for Manipulation Research

Aaron Edsinger-Gonzales and Jeff Weber

MIT Computer Science and Artificial Intelligence Laboratory
Cambridge, Massachusetts, 02139, USA
{*edsinger, jaweber*}*@csail.mit.edu*

Received (31 May 2004)

Humanoid robots found in research and commercial use today typically lack the ability to operate in unstructured and unknown environments. Force sensing and compliance at each robot joint can allow the robot to safely act in these environments. However, these features can be difficult to incorporate into robot designs. We present a new force sensing and compliant humanoid under development in the Humanoid Robotics Group at MIT CSAIL. The robot, named *Domo*, is to be a research platform for exploring issues in general dexterous manipulation, visual perception, and learning. This project is currently in the design and development phase. In this paper we describe aspects of the design, detail proposed research directions for the robot, and illustrate how the design of humanoid robots can be informed by the desired research goals.

Keywords: dexterous manipulation, humanoid robotics.

1. Introduction

Humanoid robots found in research and commercial use today typically lack the ability to operate in unstructured and unknown environments. Force sensing and compliance at each robot joint can allow the robot to safely act in these environments. However, these features can be difficult to incorporate into robot designs. In the Humanoid Robotics Group at MIT CSAIL, we are currently developing a new force sensing and compliant humanoid named *Domo*. *Domo* is to be a research platform for exploring issues in general dexterous manipulation, visual perception, and learning. This project is currently in the design and development phase.

Domo, as pictured in Figure 1, has 29 active degrees of freedom (DOF), 58 proprioceptive sensors, and 24 tactile sensors. 22 DOF use force controlled and compliant actuators which are fundamental to our research approach to manipulation. There are two six DOF force controlled arms, two four DOF force controlled hands, a two DOF force controlled neck, and a seven DOF active vision head. The torso is not currently actuated. The real-time sensorimotor system is managed by an embedded network of five DSP controllers. The vision system includes two FireWire CCD cameras[1] and utilizes the *YARP*[2] software library for visual processing. The cognitive system runs on a small, networked cluster of PCs running the Linux operating system.

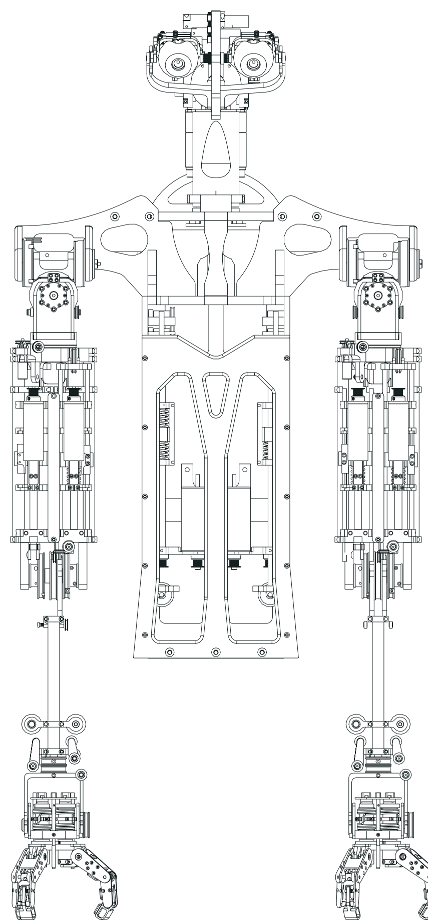


Fig. 1. *A design drawing of the robot,* Domo*, under development. The robot has 29 DOF: six in each arm, four in each hand, two in the neck, and seven in the active vision head.*

Research approach

We hope to advance a creature based approach to humanoid robotics with the *Domo* platform. A creature based robot can be left on, interacting with the environment, for extended periods of time. New behaviors can be added and integrated with existing ones. Currently, most humanoids are left to run only for the duration of an experiment. Reasons for this include a lack of electro-mechanical robustness, software architectures that lack scalability and integrability, and more generally, the nature of humanoid research which is often driven paper to paper, experiment to experiment.

A related project within our group is the Mertz platform [3], which explores a creature based approach on an active vision head identical to *Domo*'s. The long term research goal with Mertz is to explore a more scalable learning framework, inspired by how human infants learn. Mertz is to be placed in a public venue for extended periods of time, continuously interacting with different people and incrementally learning about patterns and correlations of perceived events.

Taking a creature based approach with both *Domo* and Mertz requires incorporating into the design, at all levels, the constraint that the robot needs to be able to run for hours and days without damaging itself. This approach will allow us to do longer time-scale experiments (e.g., over many days) that otherwise cannot be attempted. In Section 4, we elaborate on our proposed research directions for *Domo*.

Design approach

The design of *Domo* accommodates a creature based approach to dexterous manipulation and sensorimotor learning. These concerns require that the robot must be able to interact safely with dynamic stimuli, including humans, for extended periods of time. *Domo* also must also be able to sense and exert forces on the world. This allows the robot to operate without an accurate or complete model of its environment.

The *Domo* design is influence by previous work on the humanoid platforms, Cog[4] and Kismet[5]. While our research direction for *Domo* differs from those of Cog and Kismet, we are able to draw upon experience gained by working on these robots to build a new robot.

Humanoid robot design is particularly constrained by the adherence to an anthropomorphic form-factor. This constraint is often in opposition to the size, scale, and power density of available electromechanical systems. The mechanical complexity of humanoid machines can exceed that of automobiles. The cost of such high complexity is that the mechanical failure rate of the robot is also largely increased. Research platforms are even more prone to high failure rates due to a lack of engineering resources available to commercial platforms.

The design of such a robot's computational systems also faces difficult issues. Control of the motor systems requires precision, high bandwidth control. Sensory systems demand low latency signal acquisition. A common approach, as taken with Cog, is to physically centralize all sensorimotor and cognitive processing away from the robot. This allows for scalable computational systems and off-the-shelf sensorimotor components. However, this approach constrains the robot to be immobile (excepting the use of wireless links), and real-time access to the sensorimotor system is difficult to achieve. An embedded hardware approach allows for platform mobility and complete control of the real-time aspects of the system. It also increases the complexity of the architecture, adds development time for custom hardware, and typically exhibits a higher electrical failure rate than an off-the-shelf approach. Most embedded architectures also lack both the computational power necessary for visual perception and the ability to leverage from a preexisting code base.

We have previously experimented with placing a prototype arm design of *Domo* on a dynamically balancing base [6] to explore issues in mobile manipulation. *Domo* may eventually be mounted on a mobile platform as well. Consequently, our design is also constrained by the needs of a mobile platform, such as size, weight, and power consumption.

In the following sections we elaborate on the design of the mechanical systems, with particular attention to the force sensing actuators, arms, and hands. We then describe the architecture of the sensorimotor and cognitive systems. Finally, we discuss the research problems we hope to address with the *Domo* platform.

2. Mechanical Systems

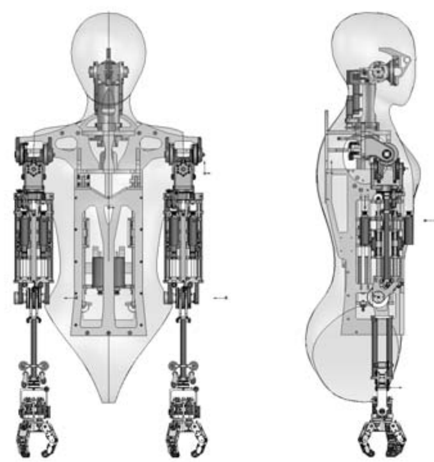


Fig. 2. *Drawing of* Domo *overlaid with silhouette of a 5′4" female. The proportions of the robot were chosen to be non-intimidating during human-robot interactions.*

The mechanical design of our robot is largely a problem of engineering. However, there are some important aspects of the design which contribute to our research program. In this section we provide an overview of the mechanical systems of *Domo* and discuss how our design considerations relate to the research goals of the platform.

A primary consideration in the design of *Domo* is the incorporation of force sensors and compliance at most of its joints. Force sensing allows us to close a control loop around the force signal, providing force control at each joint. Force control combined with natural compliance at each joint allows *Domo*'s manipulators to act safely in unknown environments. Unfortunately, off-the-shelf force sensors tend to be large and expensive. Consequently, we have designed two types of actuators for *Domo*: a new version of the Series Elastic Actuator [7] (SEA), and a novel actuator, the Force Sensing Compliant Actuator [8](FSC).

Another important design consideration is the mechanical robustness of the robot. Our previous work with Cog illustrated the difficulty in keeping a very complex machine running for extended periods of time. Mechanical failures such as faulty electrical connections and burnt out motors out can be common pitfalls. This lack of robustness limited the types of experiments that could be conducted with Cog. Keeping *Domo*'s 29 DOF running for days on end requires careful attention to modes of failure at both the hardware and software level. In Section 3 we discuss the software level modes of failure. We have dealt with mechanical modes of failure in the following ways:

- Geartrain failures: Impact shocks are commonly encountered when operating in unknown environments. These shocks can damage the spur gears in the motor's gearhead. *Domo* uses a ball screw drive combined with an elastic spring element inline with the load for each of the arm's actuators. The elastic element absorbs the high bandwidth shocks while the ball screw is less prone to damage than a standard gearhead.

- Motor winding overheating: Motor windings often burn out when experiencing prolonged stall currents. While these situations can typically be avoided in software, they are inevitable during unattended operation in unknown environments (for example, the robot's hand getting wedged underneath a table). *Domo*'s arms use custom designed brushless motor amplifiers that have built in current limit protection. Additionally, the force sensing capability of the actuators allows us to have the manipulator physically stalled while the motor is not encountering stall currents.

- Cable breakage: Cable drive systems used in *Domo* are susceptible to breaking, particularly at the termination crimps. Our design limits the direct force exerted on the crimps by either first clamping the cable or by wrapping it around a hub before termination.

- Wire strain: Electrical cables running across a moving joint encounters strain as the joint moves. Robots with centralized control and sensor hardware tend to have large, bulky bundles of wire running across joints (which also increases the load on the motor). We have alleviated this problem by providing wire routing paths through the center of rotation of the joint whenever possible. This minimizes the wire displacement during joint movement. Secondly, we have organized our control and sensor hardware in a distributed, bus configuration to minimize the number of wires that need be routed through the robot.

- Maintenance difficulty: We have designed *Domo* so that when failures *do* occur, it is easy to remove mechanical subsystems without disassembling large parts of the robot. We have taken a modular approach to the design. Actuators have a common design and can be easily removed. The arms, hands, and head can each be independently removed with ease.

A secondary design consideration is the overall size and appearance of the robot. *Domo*'s morphology is roughly analogous to that of a human. For experiments involving human-robot interaction, we believe that the size and appearance of the robot should not be intimidating to the viewer. Consequently, we have kept the proportions of the robot near that of a 5′4” female, as shown in Figure 2.

2.1. *Actuators*

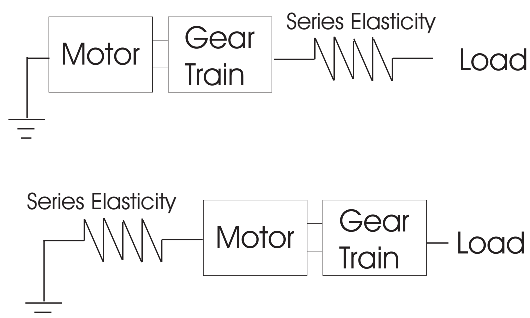
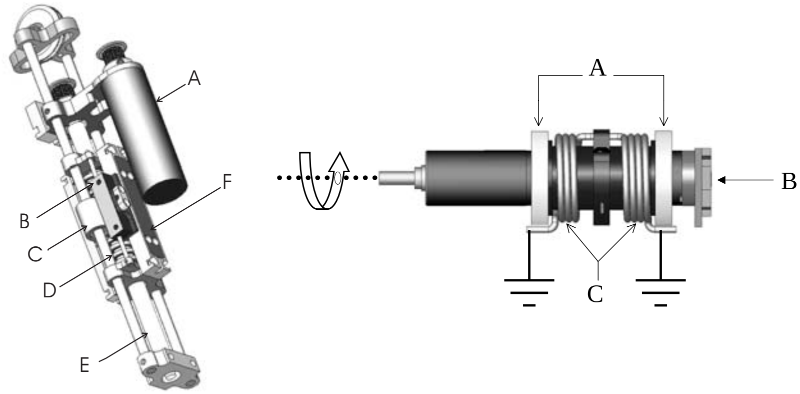


Fig. 3. *Block diagram of the Series Elastic Actuator (top) and the Force Sensing Compliant Actuator (bottom). The SEA actuator places an elastic spring element between the motor output and the load. The FSC actuator places the spring element between the motor housing and the chassis ground. SEA actuators are used in* Domo*'s arms and neck. FSC actuators are used in* Domo*'s hands.*

The 20 actuators in *Domo*'s arms and hands and the 2 actuators in the neck utilize series elasticity[7] to provide force sensing. We place a spring inline with the motor at each joint. We can measure the deflection of this spring with a potentiometer and know the force output by using Hooke's law ($F = -kx$ where k is the spring constant and x is the spring displacement). We apply this idea to two actuator configurations, as shown in Figure 3. The SEA actuator places the spring between the motor and the load, while the FSC actuator places the spring between the motor housing and the chassis ground. There are several advantages to these actuators:

- The spring and potentiometer provide a mechanically simple method of force sensing.
- Force control stability is improved when intermittent contact with hard surfaces is made. This is an important attribute for manipulation in unknown environments.



(A) SEA actuator (B) FSC actuator

Fig. 4. ***(A)*** *Model of the cable-drive SEA actuator. A brushless DC motor (A) imparts a linear motion to the inner drive carriage (C) through a precision ballscrew (E). The inner drive carriage transmits motion to the outer drive carriage (F) through two precompressed die springs (D). The deflection of the springs is measured with a linear potentiometer (B).* ***(B)*** *A simplified view of the FSC actuator. Two bearings (A) support the motor. The motor is attached to an external frame (ground) through two torsion springs (C). As the motor exerts a torque on a load, a deflection of the springs is created. This deflection is read by the torque sensing potentiometer (B).*

- Shock tolerance is improved. The use of an $N:1$ geartrain increases the reflected inertia at the motor output by N^2. This results in shock loads creating high forces on the gear teeth. The series elastic component serves as a mechanical filter of the high bandwidth forces, reducing the potential of damage to the gears.

- The dynamic effects of the motor inertia and geartrain friction can be actively cancelled by closing a control loop around the sensed force. Consequently, we can create a highly backdrivable actuator with low-grade components.

- The actuators exhibit passive compliance at high frequencies. Traditional force controlled actuators exhibit a large impedance at high frequencies because the motor response is insufficient to react at this timescale. In an SEA actuator, the impedance of the elastic element dominates at high frequencies.

The overall passive compliance exhibited by a FSC or SEA actuator is determined by the spring stiffness. If we consider that an external force applied to the actuator can only be counteracted by the spring, then we see that the mechanical impedance of the system is defined by that of the springs. The low impedance of the springs adversely affects the reaction speed, or bandwidth, of the system. For robot tasks achieved at a roughly human level bandwidth, this adverse effect is not large.

The differences between the FSC actuator and the SEA actuator provide distinct advantages and disadvantages. The SEA actuator, as pictured in Figure 4A, uses a linear ballscrew and a cable transmission. The ballscrew provides greater efficiency and shock tolerance than a gearhead. The SEA actuator is limited by the travel range of the ballscrew which creates packaging difficulties. The linear potentiometer must move with the motor output, precluding the use of continuous rotation configurations. In contrast, the FSC actuator pictured in Figure 4B can allow continuous rotation at the motor output as the potentiometer does not move with the motor output. However, the elastic element is not between the load and the geartrain, decreasing the shock tolerance.

The FSC actuator configuration used in *Domo*'s hands is highly compact due to the use of torsion springs. The packaging efficiency of torsion springs, however, does not scale well to the higher stiffness required in the arms. The torsion springs used in the hands have a stiffness of 3.85 oz-in/deg. The compression springs used in the arms have a stiffness of 300 lb/in. A torsion spring with comparable stiffness would be prohibitively large.

2.2. *Head*

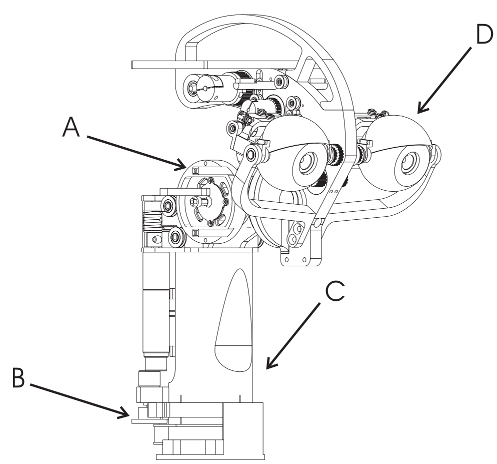


Fig. 5. *Mechanical drawing of* Domo*'s active vision head. A SEA actuator driven universal joint (B) combined with head pan (C) provides a compact ball-and-socket like three DOF neck. The upper neck provides roll and tilt through a cable-drive differential (A). Two FireWire CCD cameras (D) share a single tilt DOF and have two independent pan DOF. Expressive eyelids provide a final DOF.*

The design of *Domo*'s active vision head is an evolution from previous designs used for Cog and Kismet. It is a copy of the head used in a new active vision project by Aryananda[3] that is similarly interested in system robustness and learning during long-term interactions.

The head features seven DOF in the upper head, a two DOF force controlled neck, and a stereo pair of FireWire CCD cameras. The upper head provides roll and tilt through a compact cable-drive differential. The two cameras share a single tilt DOF and have two independent pan DOF. The head also features one DOF expressive eyelids.

The head uses brushed DC motors with both encoder and potentiometer position feedback. The potentiometers allow for absolute measurement of position at startup, eliminating the need for startup calibration routines. The analog signal from the CCD cameras is digitized to a FireWire interface on boards mounted in the head, reducing noise issues related to running the camera signals near the head's motors. Physical stops are incorporated into all DOF to safeguard the head against potential software failures. Particular attention was given to the electrical cable routing in the head. The mobility of Cog's active vision head was hampered by large cable bundles running back from the eyes through the head. *Domo*'s head features a large cable-pass down the center of the differential neck which greatly simplifies the cable routing.

Unlike the previous Cog and Kismet heads which used one wide and one foveal camera per eye, *Domo* uses a single wide angle camera per eye. The camera used is the Point Grey OEM Dragonfly[1]. The cameras are IEEE-1394 FireWire devices with 640x480 (30fps) or 1024x768 (15fps) 24-bit color resolution. We use a 2mm focal length lens. The cameras are powered by the FireWire bus and provide software based synchronization of the dual framegrabbers.

Our primary design consideration for the upper portion of the head is that it be able to execute human-like eye movement. Human eye movements can be classified as: saccades, smooth pursuit, vergence, vestibulo-ocular reflex, and the optokinetic response [9]. *Domo*'s head is designed to accommodate all but the optokinetic response. Saccades require fast, ballistic movements of the eyes ($900deg/s$) while smooth pursuit requires slow, controlled tracking movements of less than $100deg/s$. Vergence requires independent control of the eye pan to view objects of varying depth. The vestibulo-ocular reflex requires either a head mounted gyroscope or correct kinematic information (we currently use the latter). Accommodating these features required careful design of the eye drive system and motor selection. We use a small gearhead motor (Maxon 8mm 0.5W with 57:1 gearhead) and an efficient cable-drive system for the eye pan.

2.3. *Arms*

Traditional arm designs assume that end-effector stiffness and precision are necessary qualities. In the context of humanoid manipulation, however, we maintain that end-effector stiffness and precision should not be the principle consideration. Humans are notoriously bad at stiffly controlling the position of their arms but very good at controlling the forces. A central pillar of our design approach to *Domo*'s arms is that the manipulators must be passively and actively compliant and able to directly sense and command torques at each joint.

Each arm joint is driven by an SEA actuator containing a brushless DC motor.

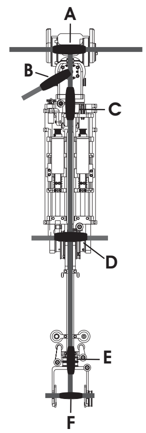


Fig. 6. *The kinematic structure of* Domo*'s arms. A compact cable-drive differential at the shoulder provides pitch (A) and roll (B). These two DOF are driven by actuators placed in the robot torso. The bicep of the arm contains four other actuators: shoulder yaw (C), elbow pitch (D), and wrist roll (E) and pitch (F) driven by two cables routed through the elbow.*

As shown in Figure 6, a compact cable-drive differential at the shoulder provides pitch and roll. These two DOF are driven by actuators placed in the robot torso. The bicep of the arm contains another four actuators for shoulder yaw, elbow pitch, wrist roll, and wrist pitch. The drive-cables for the wrist actuators are routed through the center of the elbow joint.

Domo's arms are based on a cable-drive design similar to that of the WAM arm[10]. Cable-drive systems have significant advantages. They provide higher efficiency and lower backlash than spur geartrains. They allow us to place the actuators away from the driven joints. Much of the actuator mass, which dominates the total arm mass, can be moved off of the arm or as close to the shoulder as possible. This decreases the effect of the mass on the arm dynamics during ballistic movements. It also lowers the overall energy consumption of the arm by creating a very lightweight arm which in turn can use lower wattage motors. The cable-drive design also allows us to take a modular approach to the actuator design. We achieved a more efficient and standardized packaging of the actuators than usually possible with a direct-drive approach. A typical disadvantage of a cable-drive system is that a long cable acts as a stiff spring, limiting the end effector stiffness. *Domo*'s arms, however, are not stiff due to their inherent series elasticity, making this disadvantage negligible.

Custom brushless motor amplifiers and sensory signal amplifiers are embedded throughout the arm. The physical distribution of the actuator electronics minimizes wiring run-length and simplifies cable routing. The shoulder and the wrist have hollow centers, allowing for electrical cable routing through the center of joint rotation. This increases the electrical robustness of the arm by limiting cable strain and reducing the risk of snagging the cables during movement.

High backdrivability is often desirable in humanoid arms. It decreases the susceptibility to geartrain damage and facilitates human interaction. Our experience with Cog has shown that the motors in highly backdrivable arms require greater power and are prone to heat damage because the arm must hold itself up against gravity. A manipulator with low backdrivability can hold itself up against gravity indefinitely, but is prone to geartrain damage and is cumbersome for human interaction. *Domo*'s arms are statically non-backdriveable. However, force sensing allows them to be actively backdriveable while holding static postures with only a few watts of power consumption.

2.4. *Hands*

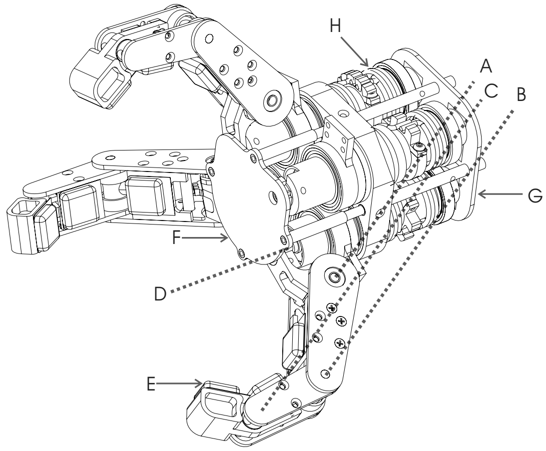


Fig. 7. *Schematic drawing of the hand: Each of three fingers has three joints (A,B,C). Joint A is driven by an FSC actuator (H) through a cable drive. Joint B is passively coupled to A through a rigid cable drive. Joint C is passively linked by a compression spring to B. The spread between two of the fingers (about axis D) is driven by FSC actuator I. The interior surface of each link in a finger has a tactile sensor (E) and the palm has an array of tactile sensors (F). Electronics for motor drive, sensor conditioning, force sensing, and controller interface reside at the rear of the hand (G).*

Hands for humanoid robots are notoriously difficult to design. Humanoid arms often impose constraints on the size, weight, and packaging of the hand while demanding sufficient dexterity, strength, and speed. These hands often lack the ability to directly sense the force applied by the actuator. They also tend to lack the mechanical robustness necessary for use in unstructured and unknown environments where impacts and collisions are common.

Humanoid hands typically use tactile sensors or load cells at the fingertips to gain force knowledge during manipulation. For example, the NASA Robonaut hand[11] utilizes Force Sensing Resistors to sense the pressure at the fingers.The Gifu Hand[12] employs a combination of load cells and tactile sensors. In either case, knowing only the fingertip forces may be insufficient when precise knowledge of the manipulating environment is not available, as is often the case in real world environments. Controllers for these hands command finger joint position. This requires accurate knowledge of the position of the manipulated object. The fingertip position must be such that the force sensor makes contact with the object for the sensor to be useful.

In contrast, a controller that can command finger joint torque is able to execute a grasp with much less accurate information. The finger need only close with a desired force and the joint position will be determined by the object being grasped. By controlling the grasp force instead grasp position, we can cast the control problem into a form that is intuitive and decomposable[13].

Each of *Domo*'s hands contains four modular FSC actuator actuators acting on three fingers, as shown in Figure 7. One actuator controls the spread between two fingers. Three actuators independently control the top knuckle of each finger. The lower knuckles of the finger are passively coupled to the top knuckle. The passive compliance of the FSC actuators is advantageous. It allows the finger to better conform to an object through local, fine-grained adjustments of posture.

The three fingers are mechanically identical, however two of the fingers can rotate about an axis perpendicular to the palm. These axes of rotation are mechanically coupled through gears, constraining the spread between the two fingers to be symmetric. By controlling the spread between two fingers, we can create a large variety of grasps, as pictured in Figure 8. Force control of the spread allows for local adjustment of grasp by simply allowing the fingers to find a local force minimum.

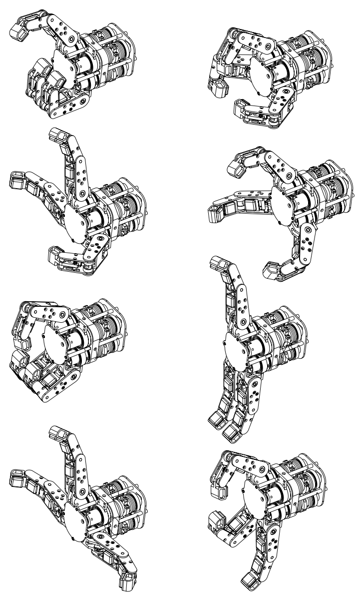
Figure 8 provides a basic design specification of the hand. The overall size, force capacity, and speed of the hand roughly conform to that of an human adult hand. We have modeled the kinematic structure after the Barrett Hand[14] which has demonstrated remarkable dexterity and grasping versatility.

3. Sensorimotor and Cognitive Systems

The design of *Domo*'s sensorimotor and cognitive system emphasizes robustness to common modes of failure, real-time control of time critical resources, and expansibility of computational capabilities. Architecturally, we draw on many of the ideas developed by Brooks[15].

The sensorimotor system includes 22 force controlled DOF, seven position controlled DOF, 58 proprioceptive sensors, 24 tactile sensors in the hands, and two cameras in the head. The cognitive system runs on an expandable networked cluster of Linux PCs.

These systems are organized into four broad layers: the *physical layer*, including sensors, motors, and interface electronics; the *DSP layer*, including real-time con-



(A)

Hand		
	Total Weight	18 oz
	Body Dim.	2.75x2.0x2.0 in
	Finger Dim.	3.66x.83x0.7 in
	Finger Tip Force	20 oz
	Curl Range	140 deg
	Spread Range	160 deg
Actuator		
	Weight	3.1 oz
	Size	1.0x1.0x2.75 in
	Torque Stall	77 oz-in
	Torque (continuous)	28 oz-in
	Speed Max	3.1 rev/s
Spring		
	Active Coils	3.25
	Diam.	0.70 in
	Wire Diam.	.0625 in
	Stiffness	3.85 oz-in/deg
	Deflection Max	20 deg

(B)

Fig. 8. ***(A)*** *The four DOF in the hand, combined with a high range of motion for each joint, provide a large variety of grasps. The finger span, when open, measures* 8.8 *inches. Each top knuckle is capable of up to* 140 *degrees of motion. The spread between the two fingers has a range of* 160 *degrees.* ***(B)*** *Design specification of the hand.*

trol; the *sensorimotor abstraction layer* providing an interface between the robot and the cognitive system; and the *cognitive layer*. The first two layers are physically embedded on the robot while the latter two are processes running on the Linux cluster. As much as possible, we have designed each layer to be robust to failures of other layers.

3.1. *Physical layer*

The *physical layer* is made up of the electromechanical resources physically embedded in the robot. This includes: 12 brushless DC motors and amplifiers in the arms, 17 brushed DC motors and amplifiers in the hands and head, a force sensing potentiometer at each of the 22 force controlled joints, a position sensing potentiometer at each of the 29 joints, a position sensing encoder at each of the 7 joints in the upper head, and an array of 12 FSR tactile sensors in each hand.

Our primary considerations in the design of the *physical layer* are the electromechanical robustness over time and the reduction of sensor noise.

Potentiometers tend to be noisy sensors. Each potentiometer sensor and FSR

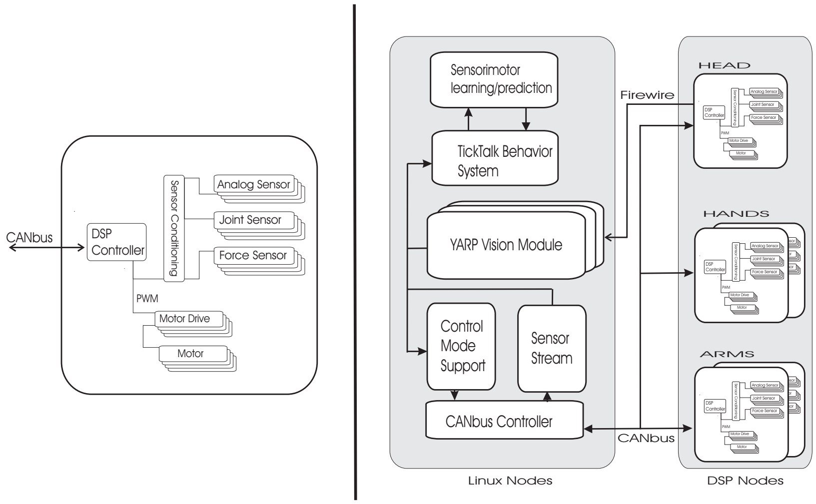


Fig. 9. *The sensorimotor and cognitive architecture of* Domo.*(Left) A DSP node handles the high bandwidth control of a set of motors and sensor acquisition. (Right) Five DSP nodes communicate on a CAN bus with the computational system. The computational system manages the sensorimotor interface, visual perception, and higher level functionality such as behaviors and learning.*

sensor on *Domo* has a signal conditioning amplifier mounted near the sensor. The amplifier provides both signal amplification and a passive low-pass filter to help reduce signal noise that may be picked up from long cable runs near the motors. The brushless motors used throughout the arms, where *Domo* encounters the longest cable runs, radiate less noise than brushed motors. In the head, the potentiometers are only sampled at startup to get the absolute position of the head for calibration, after which the encoders are used for position sensing. In the hand, we keep the cable run for the sensors as short as possible by placing the DSP controller node in the wrist.

The arm and hand motor amplifiers are embedded through out the arm (in contrast to the robot Cog, where they were kept off-board). The small form-factor custom amplifiers provide optoisolation between the motor and the controllers, 20Khz PWM switching, and current sensing and limiting. By embedding the amplifiers, we limit the noise creating inductive effect of switching high currents through long cables.

3.2. *DSP Layer*

The *DSP layer* provides real-time control over sensor signal acquisition and motor control. The layer currently incorporates five DSP nodes each running a 40Mhz Motorola DSP56F807. The nodes are mounted in the back of *Domo*'s torso. Each node communicates with the *sensorimotor abstraction layer* through a 1Mbps CAN bus channel. The number of nodes and the number of CAN bus channels is extensible as the sensorimotor requirements of the robot increase.

By using embedded DSP controllers, we gain complete control over the real-time aspects of the sensorimotor system. This avoids pitfalls commonly encountered when using PC based controllers, which can suffer from operating system timeouts and complicated startup routines. In contrast, the DSP controller starts up with a single switch and can safely function as a stand-alone unit.

Each DSP node controls up to eight joints in a 1Khz control loop. It also reads up to 16 analog sensor signals at 1Khz. The arm and hand nodes provide force control loops while the head node provides position and velocity control. Higher level controllers are implemented in the *sensorimotor abstraction layer*. In case of a failure in the *sensorimotor abstraction layer*, each node can choose to switch to a default safety mode. For example, if the CAN bus is disconnected, the arms will switch into a zero-force mode, protecting them from moving people and objects in the environment. The arms will resume activity when the CAN bus is plugged back in.

Each DSP node also performs some computation at 1Khz, such as implementing digital filters and calculating joint velocities. At a lower rate of 100hz it then communicates with the *sensorimotor abstraction layer*, relaying sensor values while accepting controller gains and commands.

3.3. *Sensorimotor abstraction layer*

The *sensorimotor abstraction layer* consists of a set of daemons running on a Linux node. It provides an interface between the *DSP layer* and the *cognitive layer*. It implements less time-critical motor controllers, interfaces with the CAN bus and the FireWire framegrabbers, and provides interprocess communication (IPC) infrastructure for the *cognitive layer*.

YARP[2] is a robot software platform developed in our lab. It enables message based IPC distributed across multiple Linux nodes. With *YARP* we can dynamically load processes and connect them into an existing set of running processes. It allows us to communicate data at a visual frame rate of $30hz$ and a sensorimotor frame rate of $100hz$.

A motor daemon interfaces with the CAN bus and polls the *DSP layer* at $100hz$ for sensory information. It implements a set of control modes available to each joint and loads the appropriate controller gains to the DSP nodes. A control mode may implement force control, joint position control, virtual spring control of joint position[13], joint velocity control, or control of the force at the end-effector. This set of control modes is made available to the *cognitive layer* through *YARP* IPC. The daemon also provides safety checks on all commands to the DSPs and places a DSP in a safety mode in case of a failure.

A vision daemon interfaces with the FireWire framegrabbers and provides visual data to the *cognitive layer* at $30fps$. It implements low-level motor controllers for the eyes, including tracking, saccades, smooth pursuit, vergence, and the vestibulo-ocular reflex. Less latency dependent mechanisms such as attention and tracking target selection are left to the *cognitive layer*.

3.4. *Cognitive layer*

The *cognitive layer* is a set of processes running on top of the *sensorimotor abstraction layer*. While much of the *cognitive layer* is the domain of future research (see Section 4), we can describe of a few of the principal components.

The visual perception system is a set of *YARP* processes which provides detection of visual features such as optic flow, color saliency, face detection, skin color, and edge-detection. This system is an extension of previous work in our group[16].

An attention system, based on Wolfe's[17] model of human visual search and attention, keeps the robot responsive to novel events and maintains an exploratory interaction with the world. Low-level visual stimuli from the visual perception system and proprioceptive and tactile sensory signals are combined to determine which environmental stimulus to attend to. Stimuli saliency is modulated by a set of motivational drives; habituation mechanisms provide a simple form of attention span for the robot.

A motivation system biases the robot to explore and manipulate its world. The motivational system maintains a homeostatic balance between a set of basic "inherent" drives and biases the *cognitive layer* to achieve these drives. With *Domo*, the motivation system can bias the visual perception toward objects that can be manipulated and that are within arm's reach.

4. Research Direction

A primary research focus of the *Domo* platform is to investigate alternative approaches to robot manipulation in unstructured and unknown environments. Today's humanoids are not able to manipulate their world at even the level of a young child. A child can grasp new objects, compensate for unknown dynamics, and explore environments with poking, pushing, touching, and grasping to learn properties of the objects in that environment. Robots certainly lack the sensory richness and the motor dexterity of a child, but we maintain that this is not the primary constraint on our ability to build capable, manipulating robots. For example, a dog can turn a bone about with two clumsy paws. Our best robots today cannot. We can build a robot with roughly the same motor dexterity as a dog (at least in terms of manipulation). We can build a robot with rich visual sensing and basic tactile sensing comparable to a dog. However, we do not yet understand how to organize the computational system that mediates between the sensory apparatus and the motor output. We lack organizing principles for building manipulating systems which can act and learn in unknown, unstructured environments.

We hope to formulate the organizing principles necessary for building robotic systems that can actively explore and learn about their environment through prolonged manipulation interactions. (Fitzpatrick[18] provides a good example of work in this direction). These principles should represent a coherent framework by which we can represent complex sensorimotor systems in a way that is conducive to learning in a generalized manipulation context.

4.1. *Learning*

It is better for *Domo* to act badly than not to act at all. Our creature based approach to *Domo* facilitates a strong bias for the robot to explore its environment and to aggressively pursue manipulation experiences. We believe that the correlations between the sensorimotor experiences generated by this exploration, combined with weak reinforcement signals generated by *Domo*'s motivation system, can provide the structural information necessary for incremental learning to occur for particular tasks. Such tasks include: learning predictive models of the sensorimotor system; performing self versus other discrimination of forces sensed by the robot; predictive preshaping of grasp based on visual features; predictive modulation of lifting and grasping force based on previous sensory experiences.

4.2. *Computation*

We are investigating novel organizations of the computation in the *cognitive layer*. Previous behavior based decompositions used on humanoid platforms, such as on Cog, tend to be coarse grained with tens and occasionally hundreds of behavioral modules. Each module is individually designed and connected to the behavioral system. We are interested in moving to a much finer grained decomposition, with potentially thousands of behavioral modules. Such an approach precludes design of the internals of each module. Instead, we would like to automatically generate the internals of the modules and the connections between them based on the long-term interactions of the robot with the environment.

To support such a fine grained approach, we plan to use the langauge TickTalk[19] currently under development at MIT CSAIL. TickTalk is a time oriented reactive language for convenient expression of parallel and communicate rich programs. It supports Lisp macros, C extensions, and very-lightweight threading, allowing thousands of real time threads. Lisp macros provide a powerful method to replicate code patterns for behaviors, while C extensions allow us to integrate TickTalk with existing machine learning and vision code bases. The TickTalk system exposes *YARP* message ports through which sensory features are updated and desired motor control commands are passed to the *sensorimotor abstraction layer*.

4.3. *Manipulation*

We are developing a behavior based approach for motor control during manipulation. This methodology is to allow for easy decomposition of manipulation problems into layered behaviors in the fashion of subsumption architecture[20]. A key component of our approach is the capacity for behaviors to rapidly switch between different controllers for subsets of the manipulator's joints. As behaviors are dynamically inhibited and subsumed during the manipulator's interaction with the environment, different controllers can rapidly engage and disengage the arm's activity. This allows for the manipulator to exhibit a rich set of control properties that are responsive to the world.

5. Conclusion

We have outlined the design and research direction of a new humanoid robot platform, *Domo*. The robot incorporates many design features which accommodate a creature based approach to humanoid robotics. This approach requires the robot to be capable of prolonged interactions with the world without danger of hurting itself or others. The robot incorporates force sensing and passive compliance at 22 of its 29 joints, allowing for safe interactions with humans and unknown environments. We have paid particular attention to mechanical and electrical robustness throughout the design. The sensorimotor and cognitive architecture for the robot provides a scalable, realtime system with safety features at multiple levels.

A creature based approach allows the robot to gain rich, prolonged sensorimotor experiences of its world during manipulation tasks. These experiences are generated from a set of core-competency behaviors and motivations. They provide the foundation for future work in sensorimotor learning and prediction.

References

1. Point Grey Research Inc, *FireFly2 IEEE-1394 CCD Camera Manual*, http://www.ptgrey.com/products/firefly2/firefly2.pdf (2004).
2. Paul Fitzpatrick and Giorgio Metta, *YARP: Yet Another Robot Platform*, MIT CSAIL, http://sourceforge.net/projects/yarp0 (2004).
3. Lijin Aryananda and Jeff Weber, MERTZ: A Quest for a Robust and Scalable Active Vision Humanoid Head Robot, in *Proceedings, IEEE-RAS International Conference on Humanoid Robotics 2004*,(Santa Monica,CA 2004, to Appear).
4. Rodney A. Brooks, Cynthia Breazeal, Matthew Marjanovic, Brian Scassellati, and Matthew M. Williamson, The Cog project: Building a humanoid robot, in *Computation for Metaphors,Analogy and Agents*, ed C. L. Nehaniv, vol. 1562 of *Springer Lecture Notes in Artificial Intelligence*, (Springer-Verlag, 1999).
5. C. Breazeal, Designing Sociable Robots (MIT Press, 2002).
6. Rodney Brooks, Lijin Aryananda, Aaron Edsinger, Paul Fitzpatrick, Charles Kemp, Una-May O'Reilly, Eduardo Torres-Jara, Paulina Varshavskaya, and Jeff Weber, Sensing and manipulating built-for-human environments, *International Journal of Humanoid Robotics*, **1**(1),(2004, to appear).
7. G. Pratt, M. Williamson, Series elastic actuators, in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-95) (Pittsburgh, PA, 1995), vol. 1 399–406.
8. Aaron Edsinger-Gonzales, Design of a compliant and force sensing hand for a humanoid robot, in *Proceedings of the International Conference on Intelligent Manipulation and Grasping* (2004, to Appear).

9. E. R. Kandel, J. H. Schwartz, and T. Jessell, *Principles of Neural Science*, McGraw-Hill and Appleton and Lange, 4th edition,(New York, NY, 2000).
10. W. T. Townsend, J. K. Salisbury, Mechanical design for whole-arm manipulation, in P. Dario, G. Sandini, P. Aebischer (eds.), Robots and biological systems: towards a new bionics? 153–164 (Springer-Verlag, 1993) .
11. C. Lovchik and M. Diftler, The robonaut hand: A dexterous robot hand for space, in *Proceedings of the IEEE International Conference on Automation and Robotics*, vol. 2, pages 907–912 (Detroit, Michigan, May 1999).
12. Kuzanuao Uchiyama Haruhisa Kawasaki, Tsuneo Komatsu, Dexterous anthropomorphic robot hand with distributed tactile sensor: Gifu hand ii, *IEEE Transactions on Mechatronics*, **7**(3):296–303 (September 2002).
13. J. E. Pratt, Virtual model control of a biped walking robot, Tech. Rep. AITR-1581, MIT Artificial Intelligence Laboratory, Cambridge, MA, USA (1995).
14. William Townsend, The barretthand grasper, *Industrial Robot: and International Journal*, **27**(3):181–188 (2000).
15. R. Brooks, L. A. Stein, Building brains for bodies, *Autonomous Robots* **1** (1994) (1) 7–25.
16. C. Breazeal, A. Edsinger, P. Fitzpatrick, B. Scassellati, and P. Varchavskaia, Social constraints on animate vision, *IEEE Intelligent Systems*, **15** (July/August 2000).
17. J.M. Wolfe, Guided search 2.0: a revised model of visual search, *Psychonomic Bulletin and Review*, **1**(3):202–238 (1994).
18. P. Fitzpatrick, From First Contact to Close Encounters: A Developmentally Deep Perceptual System for a Humanoid Robot, Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering Computer Science, Cambridge, MA (2003), available as Tech. Rep. AITR-2003-008.
19. Jonathan Bachrach, TickTalk: A Language for Sensor Networks, MIT CSAIL Research Abstracts, Cambridge, Ma, USA (2004).
20. R. A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* **2** (1986) (1) 14–23.

Aaron Edsinger received his B.S. in Electrical Engineering and Computer Science from Stanford and his M.S. in Computer Science from MIT. He works in behavior based robot manipulation with Professor Rodney Brooks as a Ph.D. candidate at MIT CSAIL.

Jeff Weber is a Research Engineer specializing in mechanical design in the Humanoid Robotics Group at MIT CSAIL.